UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/847,067	04/30/2001	Brian T. Murren	GE1-005US	4549

21718   7590   02/21/2007

LEE & HAYES PLLC
SUITE 500
421 W RIVERSIDE
SPOKANE, WA 99201

EXAMINER
SINGH, RACHNA

ART UNIT	PAPER NUMBER
2176	

SHORTENED STATUTORY PERIOD OF RESPONSE	NOTIFICATION DATE	DELIVERY MODE
3 MONTHS	02/21/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Notice of this Office communication was sent electronically on the above-indicated "Notification Date" and has a shortened statutory period for reply of 3 MONTHS from 02/21/2007.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

lhpto@leehayes.com

PTOL-90A (Rev. 10/06)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>15 June 2006</u>.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,2,4-15,17-19,30-34,37 and 38* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,2,4-15,17-19,30-34,37 and 38* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 April 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   This action is responsive to communications: Remarks filed 11/22/06.

2.   Claims 1-2, 4-15, 17-19 30-34, and 37-38 are pending. Claims 1, 11, 14, and 30 are independent claims.

Claim Rejections - 35 USC § 112

3.   Claims 37-38 rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Examiner is unable to find support for the limitations of claims 37-38. For example, the Specification does not appear to discuss the form generation procedure being independent or antecedent to a user's interaction with the computer program. Clarification is requested.

Claim Rejections - 35 USC § 102

4.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5.   Claims 1-2, 4-15, 17-19 and 30-34 are rejected under 35 U.S.C. 102(e) as being

anticipated by <u>Kougiouris et al.</u>, US 2004/0039993 A1, 2/26/04 (Filed 8/27/03,

Continuation of application filed 11/15/99).

In reference to claims 1, 15, and 17, Kougiouris teaches an automatic formatting

and validating of text for markup language graphical user interface (GUI). The GUI

markup language description comprises various types of GUI elements for which text is

to be validated and formatted such as form fields, tables, and links. See page 1,

paragraph [0010]. The GUI element may comprise one or more fields for accepting text

input and displaying text output. The markup language file GUI descriptions comprise

information usable by the validation/formatting manager component to perform various

types of validating/formatting operations. This information may exist as markup

language tag attributes, e.g., by adding custom attributes to markup language. See

page 4, paragraphs [0061]-[0064]. The user may provide text input to the GUI element

which is validated by the manager before it is displayed in HTML form. See page 5,

paragraph [0070]-[0075]. Compare to *"accessing a computer program ;*

automatically identifying a set of one or more attributes of the computer program

with values that are to be input to the computer program by a user; and

outputting an identification of the set of one or more attributes". Kougiouris further

discloses that Graphical user interfaces (GUIs) often include text fields for accepting

text input or displaying text output. For example, graphical user interfaces may comprise

a "form", that is a series of text fields with a look and feel similar to a paper-based form.

Many text fields are designed to accept text input or display text output that is often

formatted or demarcated in a particular way.   See page 1, paragraphs [0005]. The user

may provide text input to the GUI element which is validated by the manager before it is

displayed in HTML form. See page 5, paragraph [0070]-[0075]. The graphical user

interfaces can be created from markup languages such as HTML or XML-derived

markup language descriptions. Compare to ***"creating code for one or more forms***

including selected ones of the set of one or more attributes".

In reference to claim 2, Kougiouris discloses outputting the attributes in a form

such as an HTML form in which the various attributes are listed. See figures 5A-5C.

In reference to claims 4-5, Kougiouris discloses that Graphical user interfaces

(GUIs) often include text fields for accepting text input or displaying text output. For

example, graphical user interfaces may comprise a "form", that is a series of text fields

with a look and feel similar to a paper-based form. Many text fields are designed to

accept text input or display text output that is often formatted or demarcated in a

particular way.   See page 1, paragraphs [0005]. The user may provide text input to the

GUI element which is validated by the manager before it is displayed in HTML form.

See page 5, paragraph [0070]-[0075]. The graphical user interfaces can be created

from markup languages such as HTML or XML-derived markup language descriptions.

In reference to claim 6, Kougiouris discloses the user may provide text input to

the GUI element which is validated by the manager before it is displayed in HTML form.

See page 5, paragraph [0070]-[0075]. See also figures 5A-5C which illustrate a data

input field for inputting a value for the attributes. The user may also perform various

other actions causing the application to check the text, such as issuing a command to

submit the data the user has entered to a database, or perform other types of

transactions using the data. See page 8, paragraph [0125].

In reference to claims 7 and 18-19, Kougiouris teaches the user may perform

various other actions causing the application to check the text, such as issuing a

command to submit the data the user has entered to a database, or perform other types

of transactions using the data. See page 8, paragraph [0125].

In reference to claim 8, Kougiouris discloses the information may exist as markup

language tag attributes, e.g., by adding custom attributes to markup language. See

page 4, paragraphs [0061]-[0064]. The user may provide text input to the GUI element

which is validated by the manager before it is displayed in HTML form. See page 5,

paragraph [0070]-[0075].

In reference to claim 9, Kougiouris discloses outputting the attributes in a form such as an HTML form in which the various attributes are listed. See figures 5A-5C.

In reference to claim 10, Kougiouris discloses that Graphical user interfaces (GUIs) often include text fields for accepting text input or displaying text output. For example, graphical user interfaces may comprise a "form", that is a series of text fields with a look and feel similar to a paper-based form. Many text fields are designed to accept text input or display text output that is often formatted or demarcated in a particular way. See page 1, paragraphs [0005]. The user may provide text input to the GUI element which is validated by the manager before it is displayed in HTML form. See page 5, paragraph [0070]-[0075]. The graphical user interfaces can be created from markup languages such as HTML or XML-derived markup language descriptions.

In reference to claims 11-12, Kougiouris teaches the user may perform various other actions causing the application to check the text, such as issuing a command to submit the data the user has entered to a database, or perform other types of transactions using the data. See page 8, paragraph [0125]. Compare to *"identifying, for each of the command definitions of each of a plurality of interactions, the methods of the command definition"*. Kougiouris discloses the user may provide text input to the GUI element which is validated by the manager before it is displayed in HTML form. See page 5, paragraph [0070]-[0075]. See also figures 5A-5C which

illustrate a data input field for inputting a value for the attributes. The user may also

perform various other actions causing the application to check the text, such as issuing

a command to submit the data the user has entered to a database, or perform other

types of transactions using the data. See page 8, paragraph [0125]. Kougiouris

discloses outputting the attributes in a form such as an HTML form in which the various

attributes are listed. See figures 5A-5C. Compare to *"checking. . method obtains a*

value, and identifying, as an attribute. . .outputting an identification of the set of

one or more attributes".

In reference to claim 13, Kougiouris discloses GUI elements comprising user-

interface elements where the attributes are default attributes. See page 5, paragraph

[0067].

In reference to claim 14, Kougiouris teaches an automatic formatting and

validating of text for markup language graphical user interface (GUI). The GUI markup

language description comprises various types of GUI elements for which text is to be

validated and formatted such as form fields, tables, and links. See page 1, paragraph

[0010]. The GUI element may comprise one or more fields for accepting text input and

displaying text output. The markup language file GUI descriptions comprise information

usable by the validation/formatting manager component to perform various types of

validating/formatting operations. This information may exist as markup language tag

attributes, e.g., by adding custom attributes to markup language. See page 4,

paragraphs [0061]-[0064]. The user may provide text input to the GUI element which is validated by the manager before it is displayed in HTML form. See page 5, paragraph [0070]-[0075]. Compare to *"accessing a computer program; automatically identifying a set of one or more outputs of the computer program;"* Kougiouris discloses outputting the attributes in a form such as an HTML form in which the various attributes are listed. See figures 5A-5C. Compare to *"generating a list identifying the set of one or more outputs; and outputting the list"*. The said identifying one or more outputs of a computer program is *an analysis of the computer code*.

Claims 30-34 are rejected under the same rationale used in claims 1, 2, 6, 11, and 13 respectively above.

Regarding claims 37-38, Kougiouris teaches presenting a GUI with a form including text fields for accepting text input. The GUI may comprise a form that is a series of text fields with a look and feel similar to a paper-based form. See page 1, paragraph [0005].

Response to Arguments

6.   Applicant's arguments filed 06/15/06 have been fully considered but they are not persuasive. Applicant amended claims 1, 14, and 30.

With respect to claims 1 and 30, Applicant argues Kougiouris does not disclose creating **code for** one or more forms. Examiner disagrees. Kougiouris discloses that Graphical user interfaces (GUIs) often include text fields for accepting text input or displaying text output. For example, graphical user interfaces may comprise a "form", that is a series of text fields with a look and feel similar to a paper-based form. Many text fields are designed to accept text input or display text output that is often formatted or demarcated in a particular way. See page 1, paragraphs [0005]. The user may provide text input to the GUI element which is validated by the manager before it is displayed in HTML form. See page 5, paragraph [0070]-[0075]. The graphical user interfaces can be created from markup languages such as HTML or XML-derived markup language descriptions. Markup languages such as XML or HTML provide the code for creating a graphical user interface. Applicant argues the act of displaying a form does not create the code for a form. Examiner disagrees because the form is displayed using code. See page 1, paragraphs [0010]-[0011]. Furthermore, Examiner disagrees that Kougiouris's presentations already exist and does not create the code for the forms because the forms accept input from a user and display the output based on the input. In other words, after input is accepted from a user, the HTML form is generated which requires the creation of code.

With respect to claims 11, Applicant argues Kougiouris does not teach checking, for each identified method that sets a value, whether a corresponding identified method obtains the value. Examiner disagrees. Kougiouris teaches the user may perform

various other actions causing the application to check the text, such as issuing a

command to submit the data the user has entered to a database, or perform other types

of transactions using the data. See page 8, paragraph [0125]. Kougiouris discloses the

user may provide text input to the GUI element which is validated by the manager

before it is displayed in HTML form. See page 5, paragraph [0070]-[0075]. See also

figures 5A-5C which illustrate a data input field for inputting a value for the attributes.

The user may also perform various other actions causing the application to check the

text, such as issuing a command to submit the data the user has entered to a database,

or perform other types of transactions using the data. See page 8, paragraph [0125].

Kougiouris discloses outputting the attributes in a form such as an HTML form in which

the various attributes are listed. See figures 5A-5C. Compare to *"checking. . method*

obtains a value, and identifying, as an attribute. . .outputting an identification of

the set of one or more attributes". Applicant argues Kougiouris' validation operation

does not pertain to the method that sets a value, determines whether a corresponding

method obtains a value. Examiner disagrees because Kougiouris' validation operations

is determining if there was a previously set value which determines if there was a

method that previously obtained the value.

With respect to claim 14, Applicant argues there is no generation of a list

identifying a set of one or more outputs and outputting the list. Examiner respectfully

disagrees. Kougiouris discloses outputting the attributes in a form such as an HTML

form in which the various attributes are listed. See figures 5A-5C. Outputting the

attributes in a form is "outputting the list of outputs". The said identifying one or more

outputs of a computer program is *an analysis of the computer code*. Applicant

argues this is not done independent of execution of the computer program. The claim

recites "accessing a computer program; identifying a set of one or more outputs of the

computer program". It is unclear how the output of a computer program can be

analyzed without execution of the program. Clarification is requested.

In view of the comments above, the rejection is maintained.

Conclusion

7.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time

policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later

than SIX MONTHS from the mailing date of this final action.

8.   Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Rachna Singh whose telephone number is 571-272-

4099. If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Heather Herndon can be reached on 571-272-4136.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).

RS
02/06/07

Heather R. Herndon
Supervisory Patent Examiner
Technology Center 2100